Assignment 5 Report

## Each of the 5 sections explains the following for that specific algorithm:

1. How parameters were set
2. Best score
3. Run time
4. Interpretation (max/min points in a cluster, number of clusters)
5. 3 outliers based on model

**Notes:**

* Total cost demonstrates how far points are overall from their respective clusters, so a lower total cost is indicative of better clustering
* The silhouette scores represent how close points are to their respective clusters (with a maximum closeness of 1 and a minimum of -1). Thus, silhouette score means that are closer to one represent better clustering, and a goal of mine while tweaking clusters was to keep the silhouette means above 0.
* All reported scores and times may vary during different iterations of the same code; there is an element of apparent randomness within clustering so it may not match exactly.

# K-means

1. The parameter that I tweaked was k (the number of clusters). After experimenting with a large range of values, I found that setting k as 30 was the best trade-off of accuracy and efficiency
2. Total cost: 2.24 \* 106, Mean silhouette score: 0.13
3. 18.9 seconds
4. 30 clusters. **Largest:** cluster 27 had 879 points, **Smallest:** cluster 7 had 134 points.
5. Based on the cost of assigning the row of data to its cluster, three outliers were found to be rows 294, 8459, and 8121.

# K-medoids

1. Similar to k-means, k was the only parameter tweaked here. It was again determined that 30 clusters yielded solid results.
2. Total cost: 9132, Mean silhouette score: 0.094
3. 0.48 seconds
4. 30 clusters. **Largest:** cluster 14 with 718 points, **Smallest:** cluster 11 with 59 points.
5. Based on the cost of assigning the row of data to its cluster, three outliers were found to be rows 7872, 8459, 6522.

# Affinity Propagation

1. After experimenting with different values of *damp*, the dampening coefficient, I found 0.5 to be optimal. The dampening coefficient has to be between 0 and 1, and larger values increase stability of the algorithm (but slow it down in the process).
2. Mean silhouette score: 0.023
3. 47.2 seconds
4. 69 clusters (determined automatically by algorithm). **Largest:** cluster 56 with 586 points, **Smallest:** cluster 3 with 40 points.
5. Based on the lowest silhouette scores (a measure of how well a point fits into its cluster, with a higher score representing a better fit), three outliers were found to be rows 296, 271, and 63

# Hierarchical Clustering

1. Set linkage to single, which uses the minimum distance between cluster members to determine distances between clusters as a whole. Set branchorder to optimal, which orders the branches to reduce the distance between neighboring points. Also set k to 100 for this after experimentation with various options.
2. Mean silhouette score: 0.12
3. 17.2 seconds
4. 100 clusters. **Largest:** cluster 3 with 1218, **Smallest:** cluster 24 with 1
5. Based on the lowest silhouette scores (a measure of how well a point fits into its cluster, with a higher score representing a better fit), three outliers were found to be rows 8157, 6300, and 4207.

# Gaussian Mixture Model

1. I found the default parameters to work best for GMM. Most parameters, when changed, yielded lots of NaN results.
2. Average Log Likelihood: -0.34. Per documentation this “computes the average log likelihood of the GMM given all data points, further normalized by the dimension”.
3. 5.64 seconds
4. A GMM with 1 mixture, or a multivariate Gaussian. Average posterior probability that a data point belongs to the Gaussian: -2867
5. Based on average posterior probability, the outliers with the lowest probabilities of belonging to their respective clusters are rows 8051, 8459, and 6605.